

Systems of Systems and Service Oriented Architecture: Opportunities and Challenges

Presented at ESDSWG 2007

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Jim Smith
October 24, 2007



Purpose

Provide some insights into current thinking about Service Oriented Architecture (SOA), including:

- Opportunities/impediments
- Strengths/weaknesses
- Risks/rewards



Agenda

What is a SOA?

Why should I care?

Common misconceptions

What does it take to succeed with SOA?

Summary

Questions



Really Quick Overview of SOA Basics ₁

What is a SOA?

Service-oriented architecture is a way of designing systems that enables*

- Cost-efficiency
- Agility
- Adaptability
- Leverage of legacy investments



* SOA overview slides adapted (or, in some cases, just plain “borrowed”) from Lewis, G.; Morris, E.; Simanta, S.; & Wrage, L. “*Incorrect SOA Assumptions Can Lead To Mission Disaster.*” Presented at the Systems and Software Technology Conference 2007, Tampa, FL, USA, 18-21 June 2007.



Really Quick Overview of SOA Basics ₂

Services are reusable components that represent business tasks.

- Customer lookup
- Account lookup
- Credit card validation
- Weather
- Hotel reservation

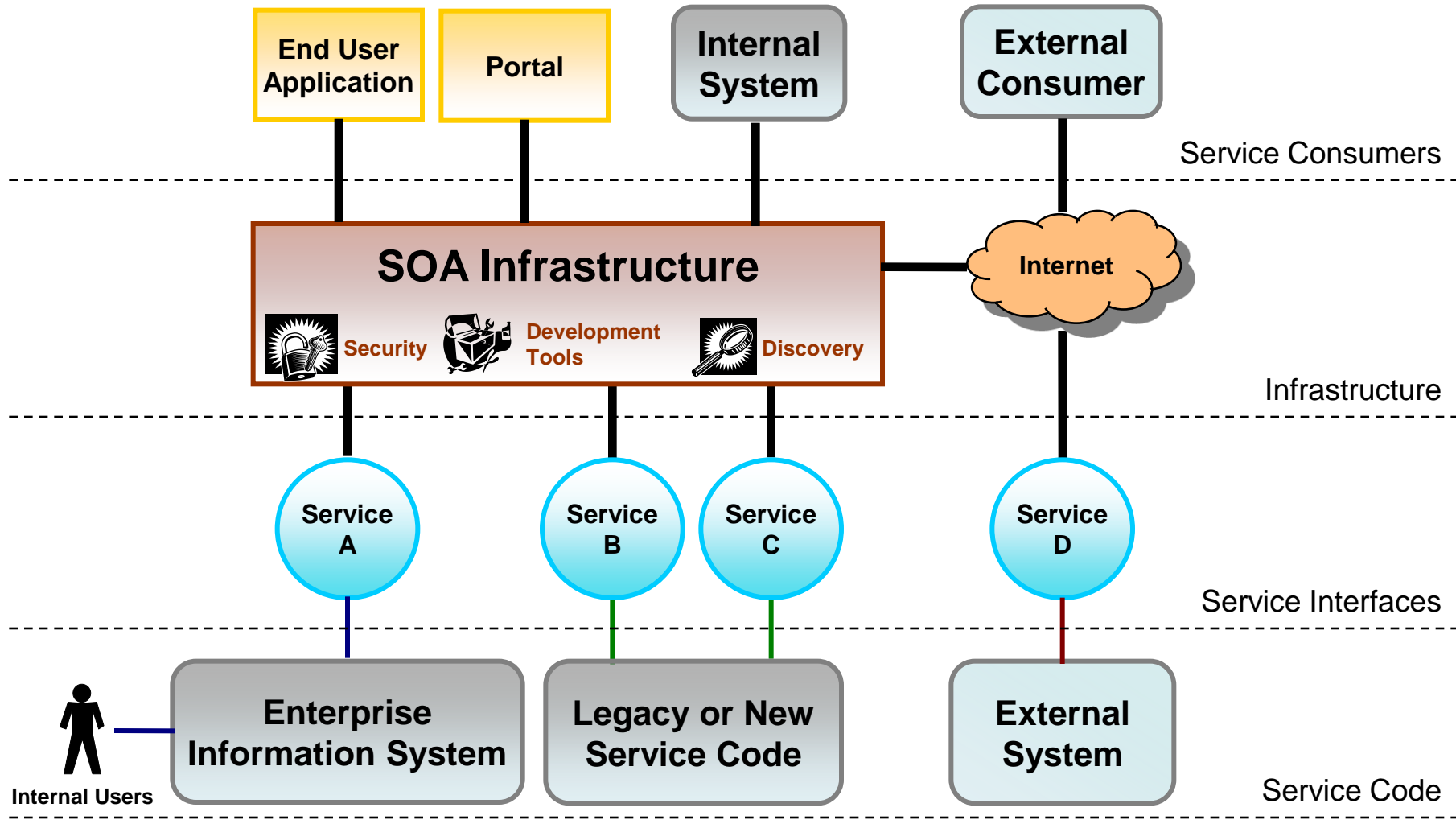


Services can be

- Globally distributed across organizations
- Reconfigured into new business processes

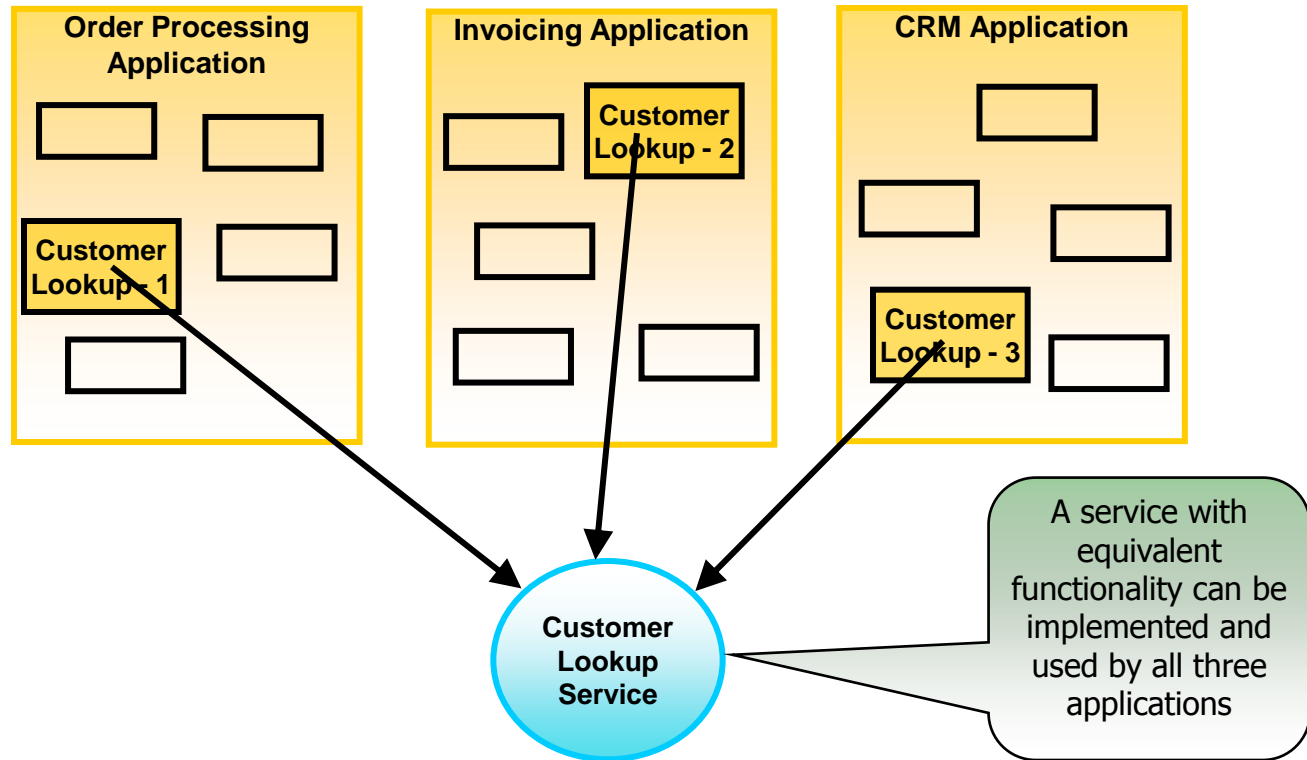


Really Quick Overview of SOA Basics ₃



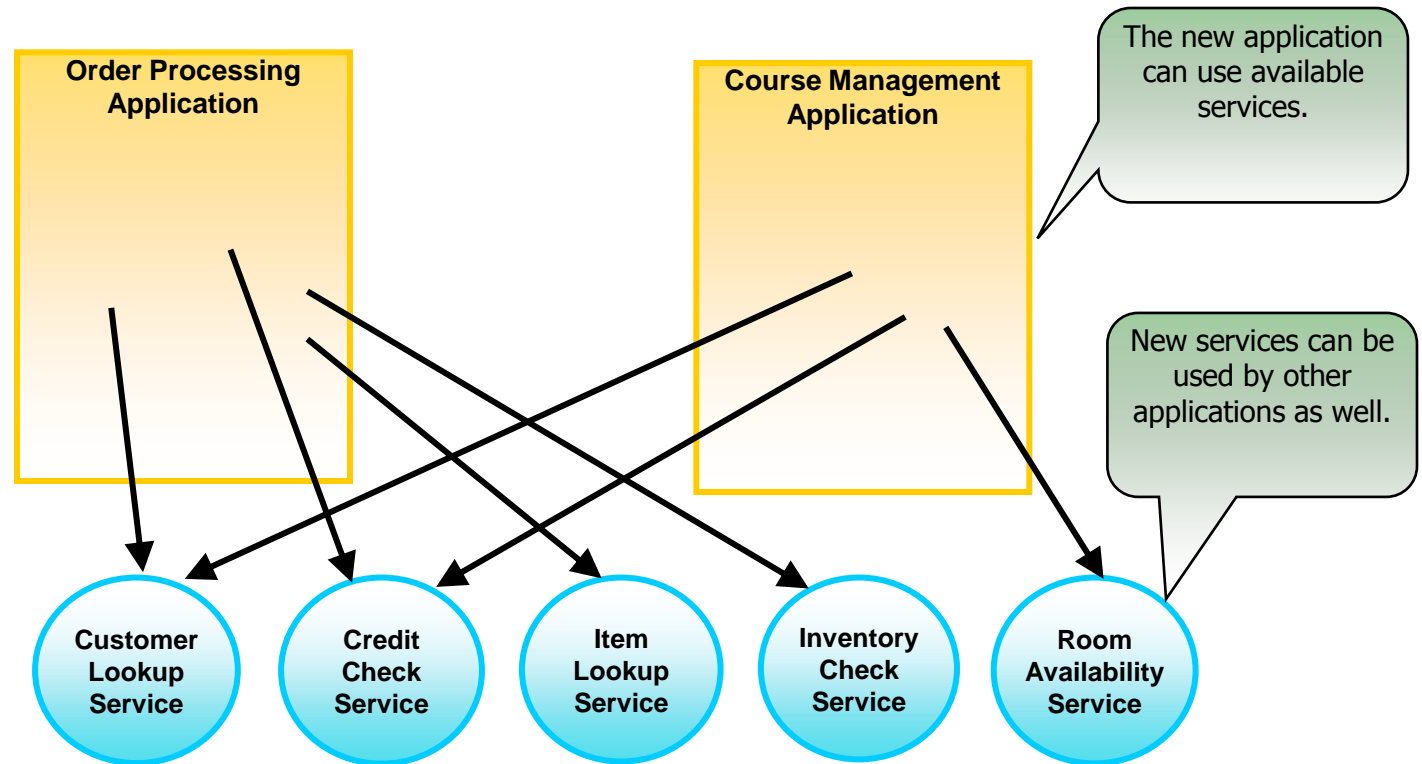
Potential Benefits From a SOA ₁

Services and Cost-Efficiency



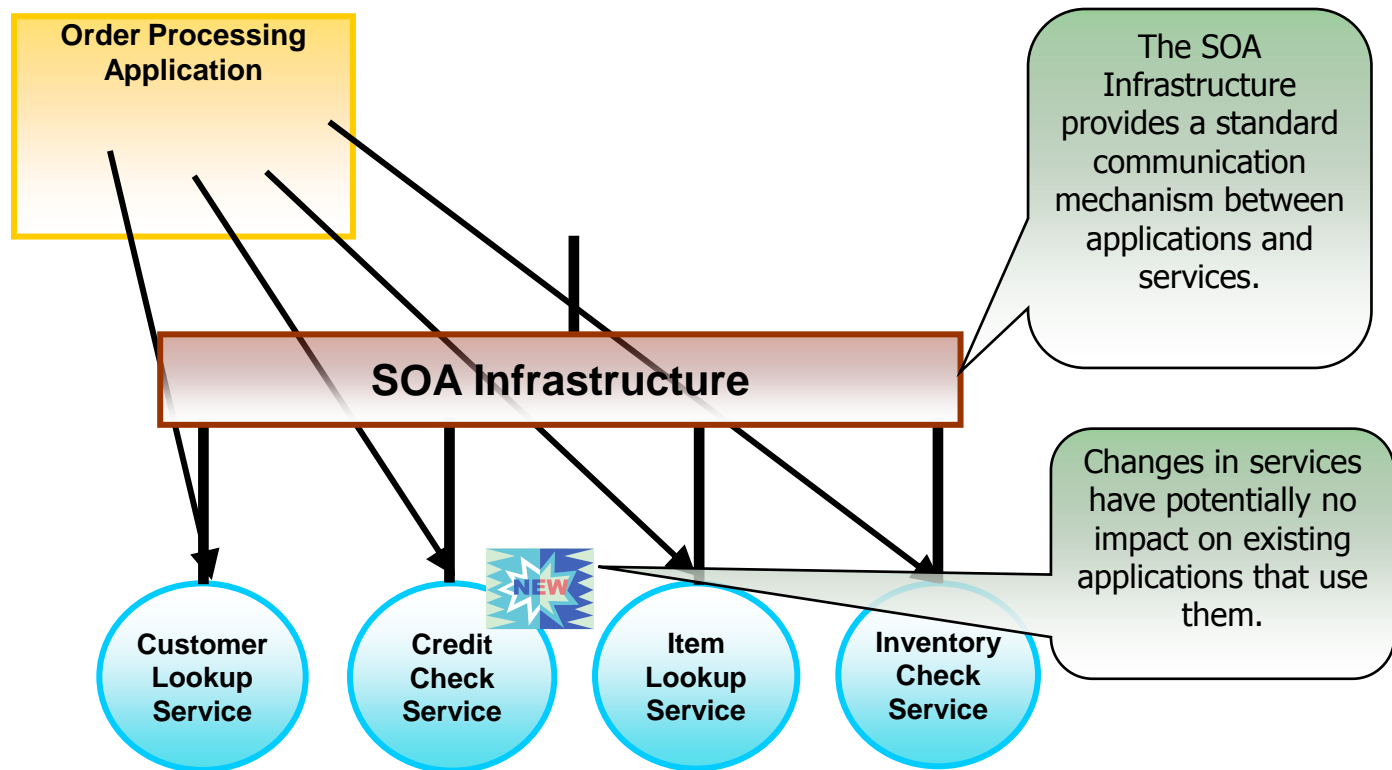
Potential Benefits From a SOA ₂

Services and Agility



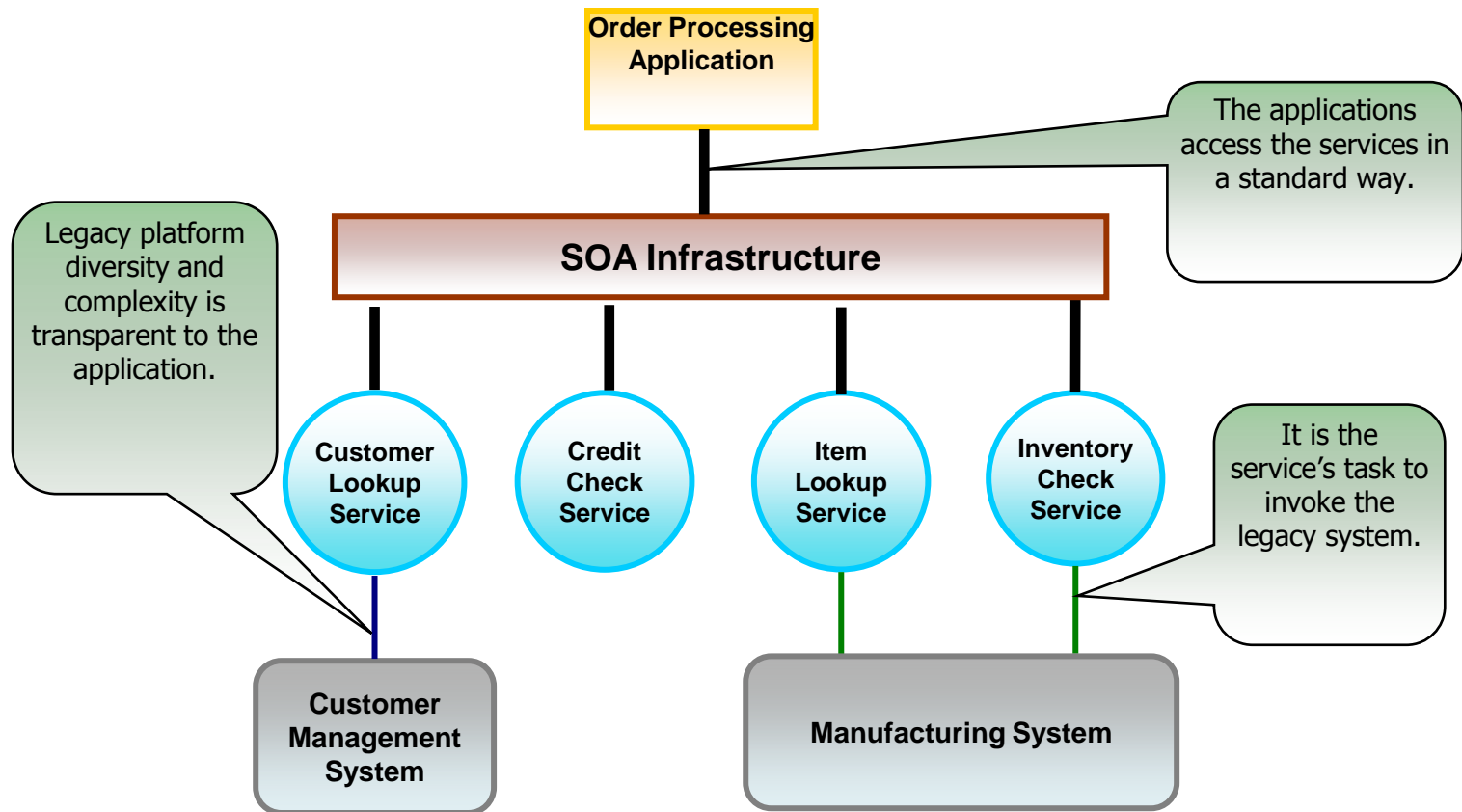
Potential Benefits From a SOA ₃

Services and Adaptability



Potential Benefits From a SOA ₄

Services and Legacy Leverage



In Summary ...

SOA is an approach to software development where

- Services provide reusable functionality with well-defined interfaces
- An SOA infrastructure enables discovery, composition and invocation of services
- Applications are built using functionality from available services

If managed well, SOA adoption can lead to

- Cost-efficiency
- Agility
- Adaptability

**However, there are some misconceptions
associated with SOA ...**



SOA Misconceptions 1

SOA is a complete architecture

- SOA is an architectural pattern ... and acts as guidance

All legacy applications can be easily migrated to an SOA

- Upfront hands-on analysis on the technical feasibility and return on investment must be performed to avoid last minute surprises: It might just not make sense to migrate the legacy system to an SOA environment

SOA Is All About Standards and Standards Are All That Is Needed

- SOA and Web Services are not the same, and most Web Services standards are emerging—subject to multiple interpretations

The Use of Standards Guarantees Interoperability in an SOA environment

- Interoperability needs agreement on both syntax and semantics
- Web Services enable syntactic interoperability, but they do not guarantee semantic interoperability



SOA Misconceptions ₂

SOA Is All About Technology

- SOA not only means a shift in technology but also changes in the organizational governance model
 - SOA governance provides a set of policies, rules, and enforcement mechanisms for developing, using and evolving SOA assets, and for analysis of their business value
 - What life-cycle model should be followed for services?
 - What other mechanisms are required?

It Is Very Easy To Develop Applications Based on Services

- It is relatively easy to build applications and services that work with a particular infrastructure ... but designing a “good” service might not be that easy

A Service Registry Allows Service Binding Dynamically at Runtime

- Current technologies have not advanced to the point that this is possible in production environments



SOA Misconceptions ₃

Testing SOA-Based Systems Is No Different than Testing Any Other Type of System

- Testing service consumers, as well as the services themselves, is challenging for various reasons:

For service consumers

- End-to-end testing is only possible when the invoked services (or test instances of them) are available
- Service consumers will necessarily have to be prepared to deal (or not to deal) with degraded service modes and complete service failure
- QoS may vary with the load on the network and on the service

For service providers

- Testing has to be based on anticipated usage patterns and scenarios
- Testing for satisfaction of SLAs is challenging
- Changes can potentially affect consumers in a negative way



SOA Misconceptions 4

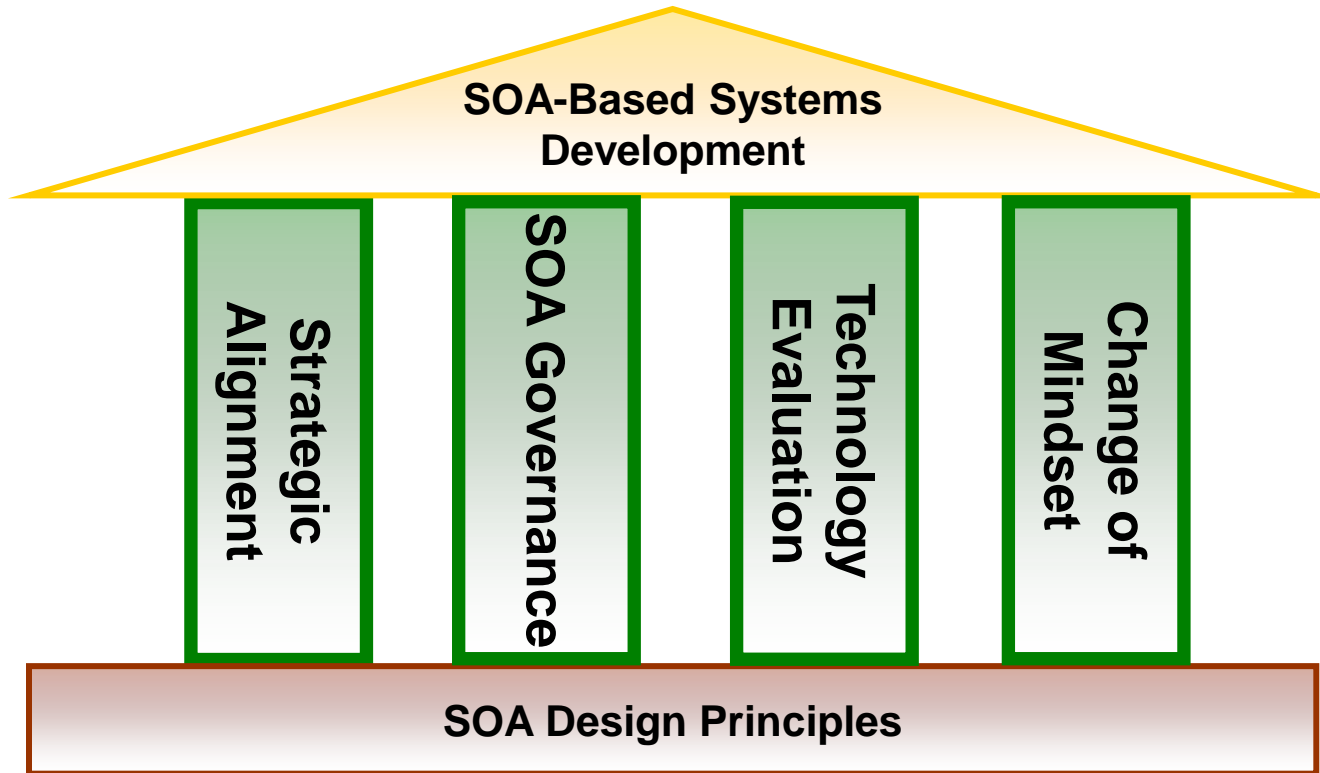
Last (But Not Least) Is The Belief That Everything In An SOA-based System Has To Be A Service

- A service-oriented approach might not make sense for the whole system.
- Guidelines for service identification are an important part of SOA strategy and governance



How To Succeed?

Pillars of SOA-Based Systems Development*



* Following slides adapted from SEI course "Migration of Legacy Components to SOA Environments," developed by Grace Lewis and Dennis Smith.

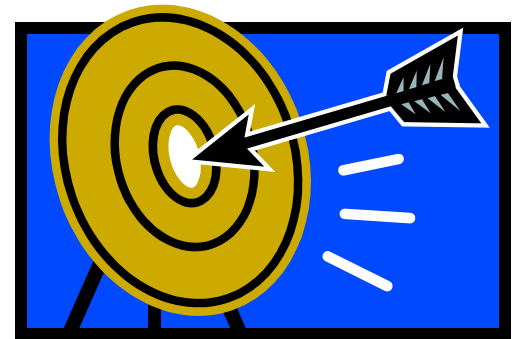


Strategic Alignment ₁

Any successful SOA strategy has to be aligned with business goals

Examples

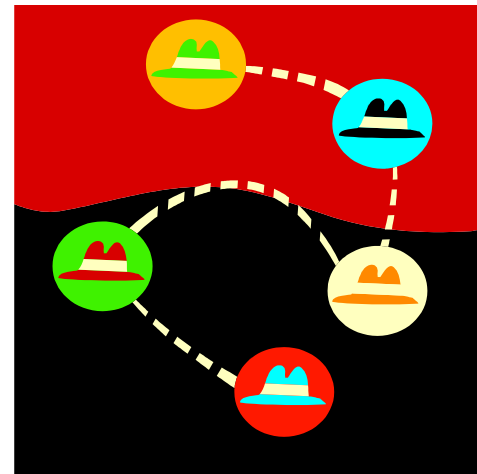
- Reduce time-to-market for applications
- Increase information available to customers
- Integrate business partners
- Decrease development cost by increasing reuse
- Reduce maintenance costs
- Improve customer service
- Improve internal processes



Strategic Alignment ₂

SOA Strategy Issues to Consider:

- Critical business goals
- Relevant business processes to support goals
 - New processes as well as processes that need to be changed
- Legacy assets
- Technology base
- Human resource base
- Business case



Strategic Alignment ₃

Linkage of Business Processes to Services

1. Business processes to support business goals are identified.
2. Candidate services are identified.

- Top-Down
 - Shared steps between business processes are identified as service candidates.



- Bottom-Up
 - Legacy system inventory is performed.
 - Systems with functionality to support business processes are identified as migration candidates.



3. Services are selected based on relationship to business goals.



Strategic Alignment 4

Lack of Governance Inhibits SOA Adoption

An InfoWorld 2007 SOA Trend Survey indicates that lack of governance is the main inhibitor for SOA adoption (50%).

- Up from 43% in 2006 Survey

Greater than other inhibitors that would seem more obvious

- Difficulty building an SOA roadmap (40%)
- Performance and reliability (39%)
- Incomplete and immature standards (39%)

Without governance, services will proliferate in a “wild west” manner.



SOA Governance ₁

SOA governance provides a set of policies, rules, and enforcement mechanisms for developing, using, and evolving SOA assets and for analysis of their business value.

- Policies and procedures
- Roles and responsibilities
- Design-time governance
- Runtime governance



SOA Governance ₂

Examples of Policies and Procedures

Service providers

- Service identification
- Service modeling
- Service development
- Service deployment
- Change management
- Service level agreements

Infrastructure providers

- Technology evaluation and selection
- Infrastructure versioning
- Upgrades
- Policy enforcement
- Monitoring

Service consumers

- Service usage
- Testing



SOA Governance ₃

Potential Role: SOA Governance Manager

Close work with

- Business process managers and analysts
- Project managers
- Infrastructure managers
- Service providers

Responsibilities

- Creation
- Approval
- Implementation
- Enforcement



SOA Governance ₄

Benefits of SOA Governance include:

- Greater alignment with business goals
- Greater control over creation, deployment, and use of services
- Centralized place for policies and regulations
- Can embed compliance with government and industry regulations
 - Sarbanes-Oxley, HIPAA, GLBA



Challenges of SOA Governance ₁

Seems counterintuitive

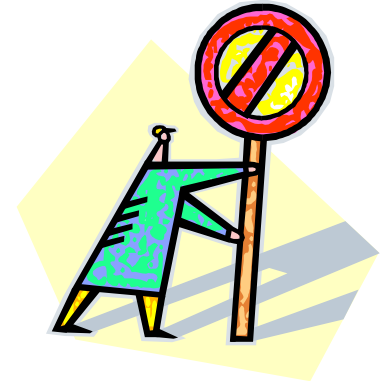
- If SOA is all about loose coupling and flexibility, why all this central control?
- Goal should be to automate governance as much as possible

Multiple organizations

- How to create governance for service providers, infrastructure providers, and service consumers?
- What if policies conflict?

Dealing with exceptions

- How to record and maintain sometimes necessary exceptions to the rules?



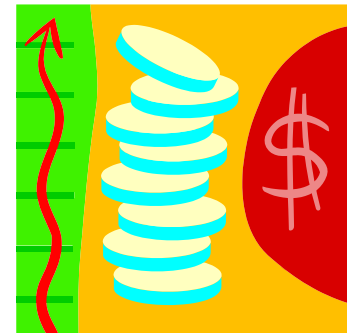
Challenges of SOA Governance ₂

Enforcing compliance

- How to make sure that policies and procedures are being followed at design time and runtime?
- What are the incentives for compliance?

Determination of ROI—needs to be factored throughout SOA life cycle

- Determine goals for each phase
- Determine desired outcomes including agility, adaptability, leverage of legacy assets, and cost savings
- Set realistic targets for timetables
- Determine operational indicators
- Measure and analyze



Technology Evaluation ₁

Match of Technologies to the Problem Domain

Need a realistic understanding on what technologies can do in the specific problem domain

How to understand and keep up with the “alphabet soup”?

- XML, SOAP, WSDL, UDDI, WS-Security?



How to determine which standards and technologies to implement in specific situations?

How to build systems that are resilient to changes in standards and commercial products that implement them?



Technology Evaluation ₂

Greatest Technology Challenge: QoS

How to determine if selected technologies will meet QoS requirements?

- Security
- Availability
- Performance

All the above questions suggest a need for **contextual experimentation**.



Different Mindset ₁

SOA-Based Systems Require a Different Development Approach

Traditional Systems Development	SOA-Based Systems Development
Tight coupling between system components	Loose coupling between consumers and services
Shared semantics at design time	Semantics ideally enable dynamic discovery and execution of services
Known set of users and usage patterns	Potentially unknown service users and usage patterns
System components all within the same organization	Multiple organizations providing system components



Different Mindset ₂

Life-Cycle Implications

Activity	Some Implications
Requirements	<ul style="list-style-type: none">• Close ties to business process modeling and analysis• May have to work with potential requirements (similar to COTS product vendors)
Architecture and Design	<ul style="list-style-type: none">• Greater emphasis on technology experimentation and tradeoff analysis
Implementation	<ul style="list-style-type: none">• Impacted by emerging standards• Requires simulation of deployment environment
Testing	<ul style="list-style-type: none">• Requires test instances for services• Greater and more diverse exception handling
Maintenance	<ul style="list-style-type: none">• Greater need for impact analysis• Greater coordination of release cycles



Different Mindset ₃

Less Control

Requires giving up full control—not easy

- Tradeoff is agility

Anticipate objections and understand validity

- Security
- Performance
- Control

Greatest challenges come from

- Single organization may not own the complete system
- Services used in unknown ways by (potentially) unknown users

Education and training on new mindset is needed.



Summary

SOAs provide a path towards a network centric model, allowing for the creation of business value in a truly distributed, dynamic fashion

Still fairly early in the “hype cycle”

Achieving greatest benefits requires careful planning and coordination across all aspects of program management, development, deployment, and sustainment

- Alignment with mission and business goals
- Instantiation of principles of SOA governance
- Evaluation of relevant technologies for SOA implementation
- Recognition that SOA requires a different mindset than traditional development



Questions?



Contact Information

Jim Smith

(703) 908-8221

jds@sei.cmu.edu

<http://www.sei.cmu.edu/staff/jds/>

ISIS Initiative

<http://www.sei.cmu.edu/isis/isis-main.html>

